International Journal of Database Theory and Application
Vol.7, No.2 (2014), pp.59-70
http://dx.doi.org/10.14257/ijdta.2014.7.2.06

An Efficient Hybrid Intrusion Detection System based on C5.0 and SVM

Vahid Golmah

Department of Computer Engineering, Neyshabur Branch, , Islamic Azad University,Neyshabur, Iran
v.golmah@in.iut.ac.ir

Abstract

Nowadays, much attention has been paid to intrusion detection system (IDS) which is closely linked to the safe use of network services. Several machine-learning paradigms including neural networks, linear genetic programming (LGP), support vector machines (SVM), Bayesian networks, multivariate adaptive regression splines (MARS) fuzzy inference systems (FISs), etc. have been investigated for the design of IDS. In this paper, we develop a hybrid method of C5.0 and SVM and investigate and evaluate the performance of our proposed method with DARPA dataset. The motivation for using the hybrid approach is to improve the accuracy of the intrusion detection system when compared to using individual SVM and individual SVM.

Keywords: *Data Mining, Intrusion Detection System(IDS), Support vectors machines (SVM), C5.0 Algorithm*

1. Introduction

The rapid development and popularity of Internet is resulted to the security of networks is increasingly become great significance and it has been a focus in the current research. Nowadays, much attention has been paid to intrusion detection system (IDS) which is closely linked to the safe use of network services. However, it is not easy to discern the attack and the normal network visit [1].

In today's intrusion detection system (IDS), large-scale data clustering and classification have become increasingly important and a challenging area. Although various tools and methods have been proposed, few are sufficient and efficient enough for real applications due to the exponential growing-in-size and high dimensional data inputs [2].

Intrusion Detection Systems (IDSs) are designed to defend computer systems from various cyber attacks and computer viruses. IDSs build effective classification models or patterns to distinguish normal behaviors from abnormal behaviors that are represented by network data. There are two primary assumptions in the research of intrusion detection: (1) user and program activities are observable by computer systems (*e.g.*, via system auditing mechanisms), and (2) normal and intrusion activities must have distinct behaviors [2].

In the intrusion detection field two different approaches can be observed: misuse detection and anomaly detection. The main idea behind misuse detection is to represent attacks in a form of a pattern or a signature in such a way that even variations of these attacks can be detected. Based on these signatures, this approach detects attacks through a large set of rules describing every known attack. The main disadvantage of the signature based approach is its difficulty for detecting unknown attacks. The main goal of the anomaly detection approach is to build a statistical model for describing normal traffic. Then, any deviation from this model

ISSN: 2005-4270 IJDTA

can be considered an anomaly, and recognized as an attack. Notice that when this approach is used, it is theoretically possible to detect unknown attacks, although in some cases, this approach can lead to a high false attack rate. This ability to detect unknown attacks has been the cause of the increasing interest in developing new techniques to build models based on normal traffic behavior in the past years [3].

The anomaly detection approach has been a very active research topic inside the machine learning community and it has been the subject of many articles over the past years. One of the most successful approaches is based on the idea of collecting data only from network normal operation. Then, based on this data describing normality, any deviation would be considered an anomaly [3].

Several machine-learning paradigms including Hidden Markov Model ,HMM [4], support vector machines, SVM [2], artificial neural networks, ANN [1],Bayesian networks, multivariate adaptive regression splines (MARS) fuzzy inference systems (FISs) [5], *etc.* have been investigated for the design of IDS. In this paper, we investigate and evaluate the performance of SVM and hybrid C5.0–SVM approach. The motivation for using the hybrid approach is to improve the accuracy of the intrusion detection system when compared to using individual approaches. The hybrid approach combines the best results from the different individual systems resulting in more accuracy. The rest of the paper is organized as follows. The Literature review is presented in Section 2 followed by a short theoretical background on Intrusion Detection Systems methods. Experimental results and analysis is presented in Section 4 and conclusions presented at the end.

2. Related Work

Different techniques and approaches have been used in later developments [6-9]. The main techniques used are statistical approaches, predictive pattern generation, expert systems, keystroke monitoring, model-based Intrusion detection, state transition analysis, pattern matching, and data mining techniques [9].

Statistical approaches compare the recent behavior of a user of a computer system with observed behavior and any significant deviation is considered as intrusion. This approach requires construction of a model for normal user behavior. Any user behavior that deviates significantly from this normal behavior is flagged as an intrusion. Intrusion detection expert system (IDES) exploited the statistical approach for the detection of intruders.

In an expert system, knowledge about a problem domain is represented by a set of rules. These rules consist of two parts, antecedent, which defines when the rule should be applied and consequent, which defines the action(s) that should be taken if its antecedent is satisfied. A rule is fired when pattern-matching techniques determine that observed data matches or satisfies the antecedent of a rule. The rules may recognize single auditable events that represent significant danger to the system by themselves, or they may recognize a sequence of events that represent an entire penetration scenario. There are some disadvantages with the expert system method. An intrusion scenario that does not trigger a rule will not be detected by the rulebased approach. Maintaining and updating a complex rule-based system can be difficult. Since the rules in the expert system have to be formulated by a security professional, the system performance would depend on the quality of the rules.

The model-based approach attempts to model intrusions at a higher level of abstraction than audit trail records. The objective is to build scenario models that represent the characteristic behavior of intrusions. This allows administrators to generate their representation of the penetration abstractly, which shifts the burden of determining what audit records are part of a suspect sequence to the expert system. This technique differs from

current rule-based expert system techniques, which simply attempt to pattern match audit records to expert rules [5].

The pattern matching approach encodes known intrusion signatures as patterns that are then matched against the audit data. Intrusion signatures are classified using structural inter relationships among the elements of the signatures. These structural interrelationships are defined over high level events or activities, which are themselves, defined in terms of low-level audit trail events. This categorization of intrusion signatures is independent of any underlying computational framework of matching. The patterned signatures are matched against the audit trails and any matched pattern can be detected as intrusion. Intrusions can be understood and characterized in terms of the structure of events needed to detect them. This system has several advantages. The system can be clearly separated into three parts, intrusion signatures as patterns, the audit trails as an abstracted event stream and the detector as a pattern matcher. This makes different solutions to be substituted for each component without changing the overall structure of the system. Pattern specifications are declarative, which means pattern representation of intrusion signatures can be specified by defining what needs to be matched than how it is matched. Declarative specification of patterns enables them to be exchanged across different operating systems with different audit trails. Intrusion signatures can be moved across sites without rewriting them as the representation of patterns is standardized. However, there are few problems in this approach. Constructing patterns from attack scenarios is a difficult problem and needs human expertise. Attack scenarios that are known and constructed into patterns by the system can only be detected. Attacks involving spoofing and passive methods of attack like wire-tapping cannot be detected [5].

The data mining approach has become an increasingly important research area in mining audit data for automated models for intrusion detection since explosive growth in databases has created a need to develop technologies that use information and knowledge intelligently. Classification, clustering, association, Regression, Sequence discovering, data visualization and prediction are usually the common methods in of building intrusion detection models is depicted. These methods are applied to audit data to compute models that accurately capture the actual behavior of intrusions as well as normal activities. The main advantage of this system is automation of data analysis through data mining, which enables it to learn rules inductively replacing manual encoding of intrusion patterns. The problem is it deals mainly with misuse detection, hence some novel attacks may not be detected [5].

Artificial neural network (ANN) is another data mining approach taken in Intrusion detection. An ANN consists of a collection of processing elements that are highly interconnected and transform a set of inputs to a set of desired outputs. Neural networks have been used both in anomaly intrusion detection as well as in misuse intrusion detection. A neural network for misuse detection is implemented in two ways. The first approach incorporates the neural network component into the existing or modified expert system. This method uses the neural network to filter the incoming data for suspicious events and forwards them to the expert system. This improves the effectiveness of the detection system. The second approach uses the neural network as a standalone misuse detection system. In this method, the neural network receives data from the network stream and analyzes it for misuse intrusion. There are several advantages to this approach. It has the ability to learn the characteristics of misuse attacks and identify instances that are unlike any which have been observed before by the network. It has a high degree of accuracy to recognize known suspicious events. Neural networks work well on noisy data. The inherent speed of neural networks is very important for real time intrusion detection. The main problem is in the training of neural networks, which is important for obtaining efficient neural networks. The training phase also requires a very large amount of data [5].

SVM are learning machines that plot the training vectors in high-dimensional feature space, labeling each vector by its class. SVMs classify data by determining a set of support vectors, which are members of the set of training inputs that outline a hyper plane in the feature space. SVM have proven to be a good candidate for intrusion detection because of their speed. SVM are scalable as they are relatively insensitive to the number of data points. Therefore the classification complexity does not depend on the dimensionality of the feature space; hence, they can potentially learn a larger set of patterns and scale better than neural networks.

Hybrid Intelligent Systems (HIS) of data mining methods offer many alternatives for unorthodox handling of realistic increasingly complex problems, involving ambiguity, uncertainty and high-dimensionality of data in Intrusion detection system. HIS are free combinations of computational intelligence techniques to solve a given problem, covering al computational phases from data normalization up to final decision making. Specifically, they mix heterogeneous fundamental views blending them into one effective working system [10]. The combined method selection is important in HIS to approve detection precise.

3. Proposed Method

A Hybrid approach is proposed for IDS, The idea behind this approach is to provide mechanisms for improving the detection precise. To introduce the proposed method, it is necessary to give a brief review of SVMs, C5.0 and thair framework intrusion detection.

3.1. SVM Classifier

SVM is developed on the principle of structural risk minimization. It is one of the learning machines that map the training patterns into the high-dimensional feature space through some nonlinear mapping. SVM has been successively applied to many applications in the multiclass classification [11]. By computing the hyper plain of a given set of training samples, a support vector machine builds up a mechanism to predict which category a new sample falls into (Figure 1).

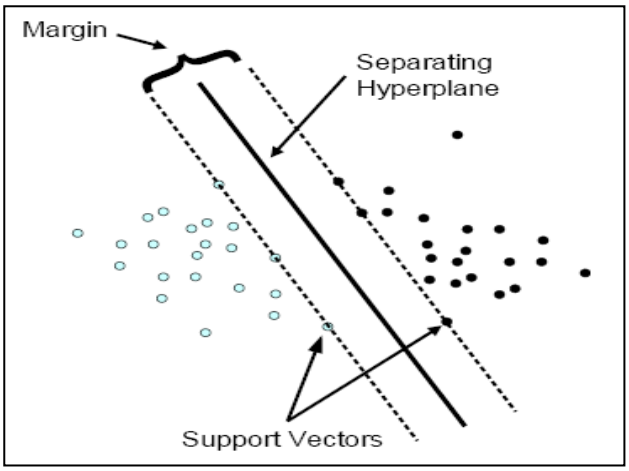

Figure 1. Separating Hyperplabe with SVM

In an SVM, a data point is viewed as a vector in the d-dimensional feature space. Assume that all data points belong to either class A or class B. Each training data point x_i can be labeled by y_i based on (1):

$$y_i = \begin{cases} -1 & x_i \in \text{classA} \\ 1 & x_i \in \text{classB} \end{cases} \quad (1)$$

Thus as it is shown in Figure 2, the training data set can be denoted as

$$D = \{(x_i, y_i) | \, i = 1, 2, 3 \ldots, N\}. \quad (2)$$

Data points with label 1 and −1 are referred to as positive and negative points, respectively. In the linear separable case, there are many hyperplanes which might separate the positive from the negative points. The algorithm simply looks for the largest margin separating hyperplane, where the ‘‘margin’’ of a separating hyperplane is defined to be the sum of the distances from the hyperplane to the closest positive and negative points (see Figure 1). In order to compute the margin of a separating hyperplane H, consider the hyperplanes H_1 and H_2 that contain the closest positive training points and the closest negative training points to H, respectively:

$$\begin{aligned} H &: w.x - b = 0, \quad x \in R^d, \\ H_1 &: w.x - b = 1, \quad x \in R^d, \\ H_2 &: w.x - b = -1, \quad x \in R^d, \end{aligned} \quad (3)$$

Where w is the normal to H and b is the distance from H to the origin. Obviously, H, H_1, and H_2 are parallel. In addition,

$$\begin{aligned} w.x_i - b &\geq 1 \text{ for } y_i = 1, \\ w.x_i - b &\leq -1 \text{ for } y_i = -1. \end{aligned} \quad (4)$$

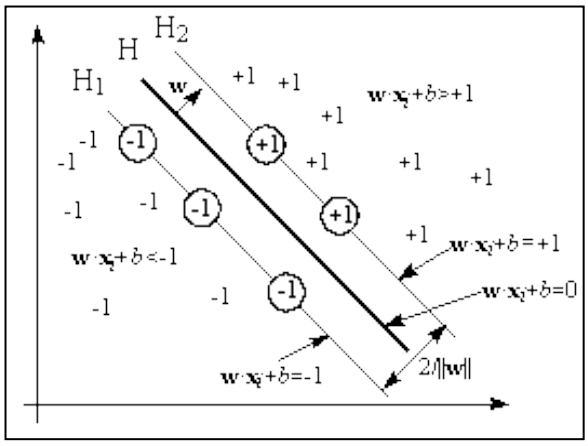

Figure 2, Data Points and Their Classes

In the case of the data points are linearly separable, above method can classify points by a linear hyperplane. But data points are nonlinearly separable, other approaches ore used for data classification such as the use of a kernel function to create a nonlinear decision boundary. A kernel function takes a data set and transforms it into a higher dimension through

the use of some function (common ones include radial basis functions, Gaussian functions, and sigmoidal functions) [1, 2, 12].

3.2. C5.0 Algorithm

Classification is an important technique in data mining, and the decision tree is the most efficient approach to classification problems. The input to a classifier is a training set of records, each of which is a tuple of attribute values tagged with a class label. A set of attribute values defines each record. A decision tree has the root and each internal node labeled with a question. The arcs emanating from each node represent each possible answer to the associated question. Each leaf node represents a predication of solution to the problem under consideration. C5.0, one of methods that be used to build a decision tree, is a commercial version of C4.5.

A C5.0 model is based on the information theory. Decision trees are built by calculating the information gain ratio. The algorithm C5.0 works by separating the sample into subsamples based on the result of a test on the value of a single feature. The specific test is selected by an information theoretic heuristic. This procedure is iterated on each of the new subsample and keeps on until a subsample cannot be separated or the partitioning tree has reached the threshold. The information gain ratio is defined as:

$$\text{Information Gain Ratio } (D, S) = \frac{\text{Gain}(D, S)}{H\left(\frac{|D_1|}{D}, \dots, \frac{|D_S|}{D}\right)} \quad (5)$$

Where in equation (5), D is a database state, H ( ) finds the amount of order in that state, when the state is separated into $S = \{D_1, D_2, \dots, D_S\}$.

The method of C5.0 is very robust for handling missing data and in a large number of input fields [13]; therefore, C5.0 is used to evaluate our features in this paper.

3.3. Hybrid Decision tree–SVM (DT– SVM) Approach

A hybrid intelligent system uses the approach of integrating different learning or decision-making models. Each learning model works in a different manner and exploits different set of features. Integrating different learning models gives better performance than the individual learning or decision-making models by reducing their individual limitations and exploiting their different mechanisms. In a hierarchical hybrid intelligent system each layer provides some new information to the higher level [5]. The overall functioning of the system depends on the correct functionality of all the layers. Figure 3 shows the architecture of the hybrid intelligent system with C5.0 and SVM. The data set is first passed through the C5.0 and node information is generated. Node information is determined according to the rules generated by the C5.0. All the data set records are assigned to one of the terminal nodes, which represent the particular class or subset. This node information (as an additional attribute) along with the original set of attributes is passed through the SVM to obtain the final output. The key idea here is to investigate whether the node information provided by the C5.0 will improve the performance of the SVM.

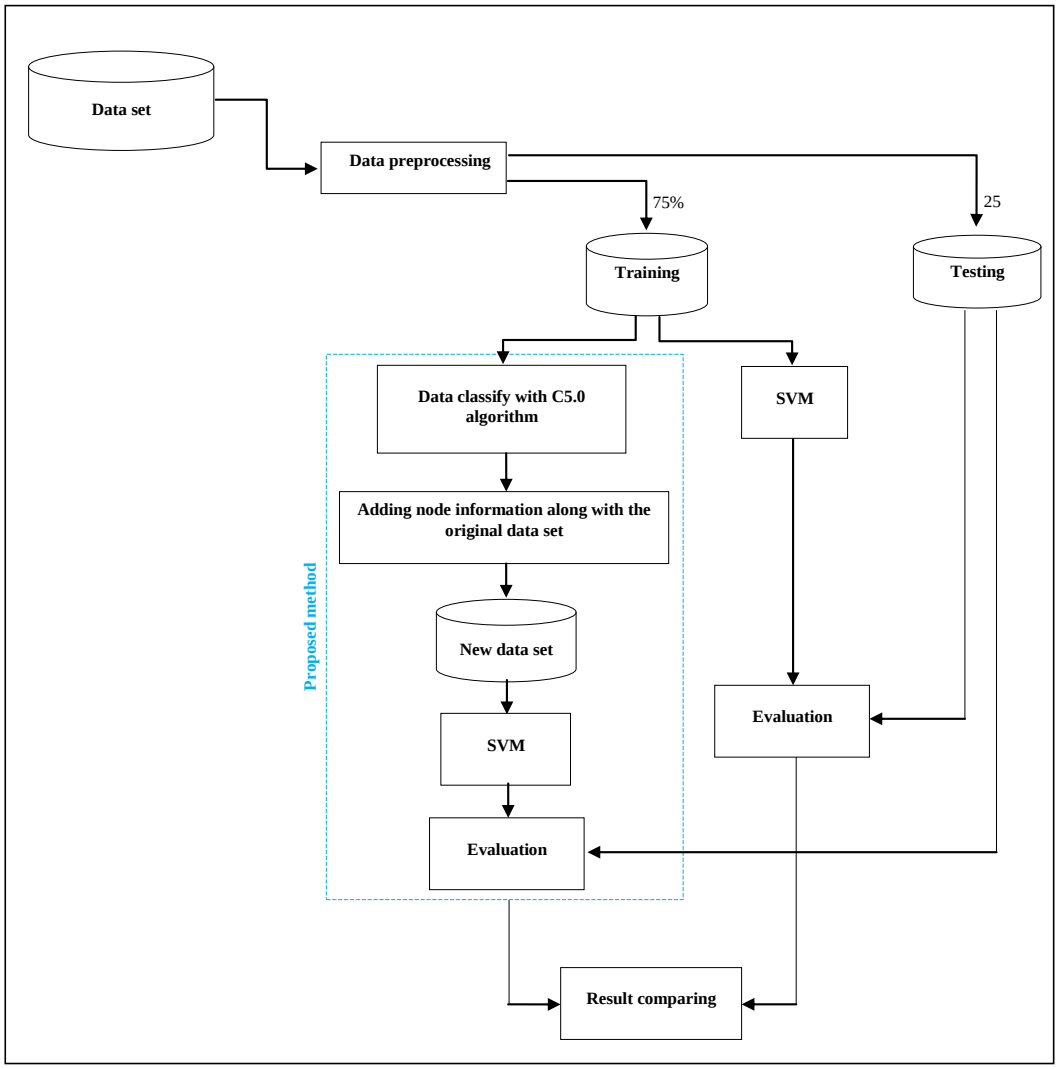

Figure 3 Architecture of the Hybrid Intelligent System with C5.0 and SVM

4. Experiments

This section evaluates the performance of the individual SVM and proposed hybrid C5.0-SVM for detection. In the individual SVM and proposed hybrid C5.0-SVM algorithms, the k-fold method is used to evaluate the accuracy of classification, and output the best test accuracy and decision rules. This study set k as 10; that is, the data was divided into 10

portions. Nine portions of data are retrieved as training data and the other one is used for testing data. In experiments, the parameter C and γ of SVM varies from 0.01 to 50,000.

4.1. Data Set and Evaluation Criteria

In this paper, the KDD Cup 1999 intrusion detection contest data is used to demo the superiority of the proposed algorithm. The KDD'99 dataset contains normal data and four types of attack: probing, denial of service (DoS), user to root (U2R), and remote to local (R2L). The four types of attack are shown in Figure 4.

In Figure 1, probing is one type of attacks that attackers can scan a network of computers to gather information or find known vulnerabilities. Denial of service is one type of attacks that attackers make some computing or memory resources too busy or too full to handle legitimate requests, or denies legitimate users access to a machine. User to root (U2R) is one type of attacks that attackers start out with access to a normal user account on the system and able to exploit system vulnerabilities to gain root access to the system. Remote to user (R2L) is one type of attacks that attackers who do not have an account on a remote machine over the network and exploit some vulnerability to gain local access as users on that machine.

There are 41 features in the KDD'99 dataset, where 32 features are continuous variable and other 9 features are nominal variable. Experiments were performed using an Intel Core 2 Duo, 2.1 GHz processor with 2 GB of RAM. We have implemented a C5.0 approach on data sets with WEKA 3.7.8.

Attack type	Attack name
Prob ing	ipsweep nmap por tswee p satan
DoS	back land neptune pod smu rf teardrop
U2R	rootkit perl loadm odu le buffer-over flo w
R2L	ftp-write spy phf guess-passwd imap warezclien t warezmaster multihop

Figure 4. The Four Types of Attack for KDD'99 Dataset

The following measurements which are often used to evaluate the efficiency of the classifier, is used in this research:

- True positive (TP_i): The number of sample that is correctly classified into the ith class;
- False positive (FP_i): The number of samples being wrongly classified into the ith class;
- True negative (TN_i): The number of outer samples that is correctly classified;

- False negative (FN_i): The number of ith class samples which is wrongly classified into the other classes;
- Precision =$\frac{\sum TP_i}{\sum TP_i+FP_i}$

On the other hand, classifier is evaluated with 10-fold cross validation, which is a technique for estimating the performance of a classifier. First, the original samples are randomly partitioned into 10 subsets. Secondly, one subset is singled out to be the testing data and the remaining 9 subsets are treated as training data. Afterwards, the cross validation process repeat 10 times and the estimation accuracy of the classifier can be evaluated by the average accuracy of the ten estimations.

The 10-fold cross validation is more popular in the circumstances of huge data set, compared with the Leave-one-out crossvalidation. The latter is usually very time expensive according to the high complexity of training times [1].

4.2. Quality of Classification

To illustrate the effectiveness of the proposed algorithm, the experiment results are shown from Table 1 and Figure 5. As shown in Table 1, the precise of classification for the proposed algorithm is 99.96%, and it outperforms individual SVM. The obtained confusion matrix resulted from methods on the training dataset are shown in Table 1 presents the 5-type classification in details: normal data, probing, DoS, U2R, and R2L.

Table 1. The Classification Precise of Classification for Various Approaches

Class	Hybrid intelligent system with C5.0 and SVM			Individual SVM [1],[14]		
	True classification ($TP_i + TN_i$)	False classification ($FP_i + FN_i$)	Precision ($\frac{\sum TP_i}{\sum TP_i+FP_i}$)	True classification ($TP_i + TN_i$)	False classification ($FP_i + FN_i$)	Precision ($\frac{\sum TP_i}{\sum TP_i+FP_i}$)
Normal	27619	147	99.47058	27319	389	98.59607
Probe	5045	0	100	5011	34	99.32607
DOS	379608	1	99.99974	379140	469	99.87645
U2R	35	0	100	31	4	88.57143
R2L	47	0	100	39	8	82.97872
Average precise	**99.96412139**			**99.78081873**		

As it is shown in Table 1, for Hybrid C5.0-SVM method there is no misclassification for probe, U2R and R2L attacks and Hybrid C5.0-SVM method can detects these attacks with 100% precise. Only one instance of DOS attack that is misclassified as normal data and 147 instances of normal that are misclassified as attack data. This means that the proposed algorithm can efficiently learn attacks with their appropriate types. Figure 5. To Compare the Precise of C5.0-SVM and Individual SVM for 5-types Attacksshows the hybrid C5.0-SVM has a high performance for 5-types attacks compared to individual SVM.

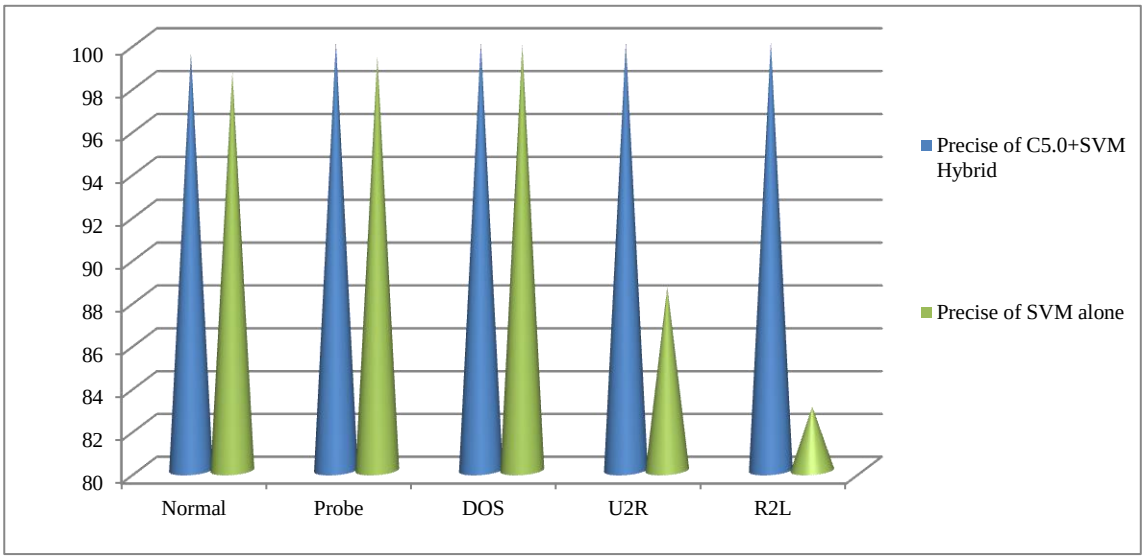

Figure 5. To Compare the Precise of C5.0-SVM and Individual SVM for 5-types Attacks

5. Conclusions

In this research, we have investigated some new techniques for intrusion detection and evaluated their performance based on the benchmark KDD Cup 99 Intrusion data. We have explored C5.0 and SVM as intrusion detection models. Next we designed a hybrid C5.0–SVM model. Empirical results reveal the hybrid C5.0–SVM approach improves or delivers equal performance for all the classes (attacks and normal) when compared to a direct SVM approach. The hybrid C5.0–SVM approach gave the best performance for probe, U2R and R2L attacks. It gives 100% accuracy for probe, U2R and R2L attacks.

References

[1] Y. Li, J. Xia, S. Zhang, J. Yan, X. Ai and K. Dai, "An efficient intrusion detection system based on support vector machines and gradually feature removal method", Expert Systems with Applications, vol. 39, **(2012)**, pp. 424-430.
[2] W. Feng, Q. Zhang, G. Hu and J. X. Huang, "Mining network data for intrusion detection through combining SVM with ant colony", Future Generation Computer Systems, (2013).
[3] C. A. Catania, F. Bromberg and C. G. Garino, "An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection", Expert Systems With Applications, vol. 39, **(2012)**, pp. 1822-1829.
[4] J. C. Badajena and C. Rout, "Incorporating Hidden Markov Model into Anomaly Detection Technique for Network Intrusion Detection", International Journal of Computer Applications, vol. 53, **(2012)**, pp. 42-47.
[5] S. Peddabachigari, A. Abraham, C. Grosan and J. Thomas, "Modeling intrusion detection system using hybrid intelligent systems", Journal of Network and Computer Applications, vol. 30, **(2007)**, pp. 114-132.
[6] H. Farhadi, M. AmirHaeri and M. Khansari, "Alert Correlation and Prediction Using Data Mining And HMM", The ISC Int'l Journal of Information Security, vol. 3, **(2011)**, pp. 77-101.
[7] L. Saganowski, M. Goncerzewicz and T. Andrysiak, "Anomaly Detection Preprocessor for SNORT IDS System", Image Processing and Communications Challenges 4: Springer, **(2013)**, pp. 225-232.
[8] X. Zhang, L. Jia, H. Shi, Z. Tang and X. Wang, "The Application of Machine Learning Methods to Intrusion Detection", Engineering and Technology (S-CET), 2012 Spring Congress on, **(2012)**, pp. 1-4.
[9] Y. Jiao, "Base on Data Mining In Intrusion Detection System Study", International Journal of Advanced Computer Science, vol. 2, **(2012)**.

[10] K. Satpute, S. Agrawal, J. Agrawal, and S. Sharma, "A Survey on Anomaly Detection in Network Intrusion Detection System Using Particle Swarm Optimization Based Machine Learning Techniques", Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA), **(2013)**, pp. 441-452.
[11] S.-W. Lin, K.-C. Ying, C.-Y. Lee and Z.-J. Lee, "An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection", Applied Soft Computing, vol. 12, **(2012)**, pp. 3285-3290.
[12] L. Wang, "Support Vector Machines: Theory and Applications", Berlin: Springer-Verlag Berlin Heidelberg, **(2005)**.
[13] A. Suebsing and N. Hiransakolwong, "Euclidean-based Feature Selection for Network Intrusion Detection", in International Conference on Machine Learning and Computing. vol. 3 Singapore: IACSIT Press, **(2009)**, pp. 222-229.
[14] W. li and Z. Liu, "A method of SVM with Normalization in Intrusion Detection", Procedia Environmental Sciences, vol. 11, **(2011)**, pp. 256-262.

